Documentation

Used Eclipse as an IDE.

Setting some project facets; Java (min SDK 11), Spring Framework (Spring Boot)

Maven added

I have been signed up to MongoDB Cloud Atlas just for this study. (Created DB, Collections, Capped, Documents, Fields) Non-Relational Database

Logging slf4j added

application.properties definitions had been done.

• Will update stock records(MongoDB capped collection is enough queue like FIFO).

(Hint: what happens if 2 or more users try to buy one last book at the same time)

In this case, we will only consider;

• Registering New Customer (/saveBook) @PostMapping It can take DtoBook dto as a parameter. Dummy data given in comment. They would not register via the same email again and again.

• Placing a new order (/saveOrder) dto.getAmount()>0 so you can not buy -1 items.

• Tracking the stock of books (/booksInStock)

• List all orders of the customer(/ordersOfCustomer/{customerId}) @GetMapping It can take two parameters as startDate and endDate.

• Viewing the order details(/viewOrderDetail) LoggerFactory.*getLogger* Result shown on the log.

• Query Monthly Statistics(/monthlyOrder) stream().filter/sorted Result shown on the console

**Be sure our system is error-proof enought.**

* *isNoneBlank* (Checks if none of the CharSequences are empty (""), null or whitespace only)
* Objects.*nonNull*(dto.getBookId()) (Returns true if the provided reference is non-null otherwise returns false)
* BeanUtils.*copyProperties(*Will clone copy of the objects)

**Postman**

* GET

{ "customerId" : 3 }

<http://localhost:7070/ordersOfCustomer/3>

With no parameters

http://localhost:7070/allOrder

* POST

{ "name": "Volkan",

"surName": "Keles",

"email": "volkykls@hotmail.com",

"statu": 0 }

<http://localhost:7070/saveCustomer>